I CLAIM:

1. Apparatus for processing data, said apparatus comprising:

an instruction pipeline operable to hold a plurality of program instructions at respective different stages of instruction processing, an execution stage of said instruction pipeline serving to execute a current program instruction held at said execution stage;

an instruction prefetch unit operable to fetch program instructions from a memory system to said instruction pipeline; and

an exception controller responsive to an exception signal to trigger exception processing by forcing program execution starting from an exception handling program instruction stored at a predetermined memory location; wherein

upon receipt of said exception signal part way through execution of said current program instruction, said exception controller is operable to trigger said instruction prefetch controller to start fetching of said exception handling program instruction from said memory system prior to completion of execution of said current program instruction.

2. Apparatus as claimed in claim 1, wherein execution of said current instruction lasts for a plurality of clock cycles and fetching of said exception handling program instruction starts part way through said plurality of clock cycles.

3. Apparatus as claimed in claim 1, wherein said exception handling program instruction redirects program execution to an exception handling routine.

4. Apparatus as claimed in claim 1, wherein said memory system comprises a cache memory and a main memory, said prefetch controller triggering a cache line fill operation upon a miss of a lookup for said exception handling program instruction within said cache memory.

5. Apparatus as claimed in claim 1, wherein said exception controller is an interrupt controller, said exception signal is an interrupt signal and said exception handling program instruction is and interrupt handling program instruction.

6. Apparatus as claimed in claim 1, wherein said exception is one of a data abort and a prefetch abort.

7. Apparatus as claimed in claim 1, wherein said instruction pipeline, said instruction prefetch unit and said exception controller are parts of a processor core.

8. Apparatus as claimed in claim 1, wherein said apparatus is an integrated circuit.

9. Apparatus for processing data, said apparatus comprising:

a cache memory operable to store program instructions to be executed; and

an exception controller responsive to an exception signal to trigger exception processing by forcing program execution starting from an exception handling program instruction stored at a predetermined memory location; wherein

upon receipt of said exception signal part way through execution of a current program instruction, said exception controller is operable to trigger a lookup of said exception handling program instruction within said cache memory and if said exception handling program instruction is not present within said cache memory to trigger a cache linefill operation to read said exception handing program instruction from a main memory to said cache memory.

10. Apparatus as claimed in claim 9, wherein execution of said current instruction lasts for a plurality of clock cycles and lookup of said exception handling program instruction within said cache memory starts part way through said plurality of clock cycles.

11. Apparatus as claimed in claim 9, wherein said exception handling program instruction redirects program execution to an exception handling routine.

12. Apparatus as claimed in claim 9, wherein said exception controller is an interrupt controller, said exception signal is an interrupt signal and said exception handling program instruction is and interrupt handling program instruction.

13. Apparatus as claimed in claim 9, wherein said exception is one of a data abort and a prefetch abort.

14. Apparatus as claimed in claim 9, wherein said cache memory and said exception controller are parts of a processor core.

15. Apparatus as claimed in claim 9, wherein said apparatus is an integrated circuit.

16. A method of processing data, said method comprising the steps of:

executing a current program instruction held at an execution stage of an instruction pipeline;

fetching program instructions from a memory system to said instruction pipeline; and

in response to an exception signal, triggering exception processing by forcing program execution starting from an exception handling program instruction stored at a predetermined memory location; wherein

upon receipt of said exception signal part way through execution of said current program instruction, starting fetching of said exception handling program instruction from said memory system prior to completion of execution of said current program instruction.

17. A method as claimed in claim 16, wherein execution of said current instruction lasts for a plurality of clock cycles and fetching of said exception handling program instruction starts part way through said plurality of clock cycles.

18. A method as claimed in claim 16, wherein said exception handling program instruction redirects program execution to an exception handling routine.

19. A method as claimed in claim 16, wherein said memory system comprises a cache memory and a main memory, a cache line fill operation being triggered upon a miss of a lookup for said exception handling program instruction within said cache memory.

20. A method as claimed in claim 16, wherein said exception is an interrupt, said exception signal is an interrupt signal and said exception handling program instruction is and interrupt handling program instruction.

21. A method as claimed in claim 16, wherein said exception is one of a data abort and a prefetch abort.

22. A method as claimed in claim 16, wherein said method is performed within a processor core.

23. A method as claimed in claim 16, wherein said method is performed within an integrated circuit.

24. A method of processing data, said method comprising the steps of:

storing program instructions to be executed within a cache memory; and

in response to an exception signal, triggering exception processing by forcing program execution starting from an exception handling program instruction stored at a predetermined memory location; wherein

upon receipt of said exception signal part way through execution of a current program instruction, triggering a lookup of said exception handling program instruction within said cache memory and if said exception handling program instruction is not present within said cache memory to trigger a cache linefill operation to read said exception handing program instruction from a main memory to said cache memory.

25. A method as claimed in claim 24, wherein execution of said current instruction lasts for a plurality of clock cycles and lookup of said exception handling program instruction within said cache memory starts part way through said plurality of clock cycles.

26. A method as claimed in claim 24, wherein said exception handling program instruction redirects program execution to an exception handling routine.

27. A method as claimed in claim 24, wherein said exception signal is an interrupt signal and said exception handling program instruction is and interrupt handling program instruction.

28. A method as claimed in claim 24, wherein said exception is one of a data abort and a prefetch abort.

29. A method as claimed in claim 24, wherein said method is performed within a processor core.

30. A method as claimed in claim 24, wherein method is performed within an integrated circuit.